Andrew Yi

☐ (917) 246-7405 | ✉ andrewyi@gatech.edu | ⬡ andrewhyi | ⬛ andrewhyi

Education

Georgia Institute of Technology   *Atlanta, GA*

MS IN COMPUTER SCIENCE, GPA: 4.0/4.0   *Aug 2019*

Syracuse University   *Syracuse, NY*

BS IN ECONOMICS, GPA: 3.48/4.0   *Dec 2008*

Experience

Riley & Grey   *Brooklyn, NY*

TECHNICAL LEAD   *Jun 2017 - Apr 2019*

- Developed with and led a team of three developers and two designers on features involving website template design, back-end services development/maintenance (API, payments, DevOps), website hosting, CI/testing/deployment and bug support (Ruby on Rails, JavaScript, Heroku).
- Architected, developed and successfully launched the company's first physical product – a template-based paper stationery printing service, featuring an in-browser stationery editor, proofing/shipping/e-commerce systems and back-end integration with paper printing provider (Ruby on Rails, ReactJS).
- Developed internal customer support tools to automate various time-consuming support related tasks, foregoing need to hire additional customer support agents.
- Maintained, patched and upgraded all services/systems used in the company's technology stack (Ruby, Ubuntu, PostgreSQL, Redis, etc.).

SOFTWARE ENGINEER   *Jan 2015 - May 2017*

- Developed company's core products: website hosting, domain provisioning, website templates, and RSVP/guest management.
- Implemented 30+ website design templates, working closely with both design and front-end development teams, with an emphasis on mobile responsive design, accessibility and design documentation.
- Refactored legacy CSS code to be more maintainable and modular, improving testability, reducing front-end regression bugs and increasing design/development iteration speeds (SCSS, BEMS, Selenium, testing).
- Reduced server costs by 50% by minimizing server memory bloat, optimizing SQL queries and adding a caching layer to better handle high-traffic websites (Memcached).
- Spearheaded efforts to migrate legacy AngularJS (1.x) code to modern ReactJS, with a focus on augmenting existing test suite with automated browser tests.

Friends & Family Foundry   *Brooklyn, NY*

SOFTWARE ENGINEER (CONTRACT)   *Oct 2018 - May 2019*

- BUILDBOOK - CONSTRUCTION MANAGEMENT MOBILE APPLICATION (app.buildbook.co): Led team of three developers, architecting and developing RESTful JSON API (Ruby on Rails) and iOS/Android applications (React Native) featuring: payments, photo uploading, message/media caching, push notifications, full-text search (PostgreSQL) and project management application endpoints. Developed using Test-Driven Development (TDD, RSpec, Swagger) and Agile methodologies.
- Successfully launched and published MVP iOS application (NAHB International Builders Show, Las Vegas 2019).

SOFTWARE ENGINEER   *Jul 2013 - Dec 2014*

- Built various web application back-end and front-end features for clients including Starbucks (store.starbucks.com) and IVY (ivy.com) (Ruby on Rails, AngularJS, ReactJS).

Parcel (thisisparcel.com)   *New York, NY*

FOUNDING ENGINEER   *Jan 2017 - Apr 2019*

- Developed an integrated suite of applications including a Chrome browser extension and Web/iOS applications focused on shopping, featuring product bookmarking, website scraping, price tracking and budgeting (Ruby on Rails, AngularJS, Selenium, Web Scraping).

Projects

Lunar Lander (OpenAI) (https://gitlab.com/AndrewHYi/rl-lunarlander-showcase**)**

CS 7642 REINFORCEMENT LEARNING

- Trained agents using the Double Q-learning algorithm and soft target update rule (Python, TensorFlow/Keras, OpenAI Gym).

Sleep Stage Classification (https://gitlab.com/AndrewHYi/bd4h-sleepstage-showcase**)**

CS 6250 BIG DATA FOR HEALTH INFORMATICS

- Implemented and trained CNNs for single-channel EEG sleep stage classification using the National Sleep Research Institute's Sleep Heart Health Study datasets (Python, TensorFlow/Keras, NumPy, Scikit-learn, Hadoop).

Stereo Correspondence (https://gitlab.com/AndrewHYi/cv-stereo-showcase**)**

CS 6476 COMPUTER VISION

- Implemented and analyzed window-based stereo correspondence techniques, including the sums of squared differences measure and loopy belief propagation (Markov random fields) using the Middlebury Stereo datasets (Python, NumPy).